# R語言

## 1.3

1. Assign value : ① x=7 or x<-7
   ② can be overwritten eg: x<-7, x<-9 ⇒ 則x變成9
2. ls( ) : let us know everything that stored in the workspace memory
   eg: ls( ) ⇒ x, y
3. rm( ) : remove an object from the workspace memory
   eg: rm(y) ⇒ y 就不見了!
4. 可四則運算.
   sqrt( ) : = y^(1/2) : 開根號.
   log( )
   abs( ) : 絕對值.
5. 井... : 忽略.

## 1.4 Creat and Work with Matrics

1. c( ) : concatenate 連結起來 → 形成向量 vector
   ① x1 <- c(1,3,5,7,9) ⇒ x1 ⇒ 1 3 5 7 9
   ② gender <- c("male", "female") ⇒ [1] "male" "female"
2. 2:7 ⇒ 2 3 4 5 6 7
3. seq(from=1, to=7, by=1) ⇒ [1] 1 2 3 4 5 6 7
4. rep(1, times=5) ⇒ [1] 1 1 1 1 1 or ② rep("marin", times=3) ⇒ [1] "marin" "marin" "marin"
   ③ rep(seq(from=2, to=5, by=0.25), times=5)
5. 可對向量做四則運算, 如果兩個向量長度一樣, 也可做四則運算.
6. 可取其中幾個分量.
   eg: y<-c(1,3,5,7,9)
   ① y[3] ⇒ 5 : 取第3個分量.
   ② y[-3] ⇒ 1 3 7 9 : 除了第3個以外
   ③ y[1:3] ⇒ 1 3 5 : 取第1~3個
   ④ y[c(1,5)] ⇒ 1 9 : 取第1&5個.
   ⑤ y[-c(1,5)] ⇒ 3 5 7 : 取第1&5以外
   ⑥ y[y<6] ⇒ 1 3 5 : 取小於6的值.

7. mat <- matrix(c(1,2,3,4,5,6,7,8,9), nrow=3, byrow=TRUE).

① ↳

| | [,1] | [,2] | [,3] |
|---|---|---|---|
| [1,] | 1 | 2 | 3 |
| [2,] | 4 | 5 | 6 |
| [3,] | 7 | 8 | 9 |

② if byrow=FALSE

| | [,1] | [,2] | [,3] |
|---|---|---|---|
| [1,] | 1 | 4 | 7 |
| [2,] | 2 | 5 | 8 |
| [3,] | 3 | 6 | 9 |

8. 取出 matrix 中的分量.

① mat[1,2] ⇒ 2

② mat[c(1,3),2] ⇒ 2 8

③ mat[2,] ⇒ 4 5 6

④ mat[,1] ⇒ 1 4 7

⑤ mat*10 ⇒

| | [,1] | [,2] | [,3] |
|---|---|---|---|
| [1,] | 10 | 20 | 30 |
| [2,] | 40 | 50 | 60 |
| [3,] | 70 | 80 | 90 |

## 1.5a Import Data, Copy Data from Excel to R (.csv and .txt files)

1. ①.csv ⇒ 用，隔開

②.txt ⇒ 用 tab 隔開

2. 先把 excel 中的資料存成 csv 或 txt 檔.

csv.

[Way 1]

3. data1 <- read.csv(file.choose(), header=TRUE)
   ↳可寫T.
   ↳(第一列是不是變數名稱).

[Way2]

4. data2 <- read.table(file.choose(), hearder=T, sep=",")

txt

[Way1]

5. data3 <- read.delim(file.choose(), header=T)

[Way2]

6. data4 <- read.table(file.choose(), header=T, sep="\t")

## 1.5b Import/Read Excel Data into R using readxl built-in Package & Menu

The readxl package can import both .xlsx and .xls files (this package is pre-installed

[Way3]

①

1. File → Import Dataset → From Excel

or ② 右上角 Import Dataset → From Excel → (出現視窗，可設定要輸入的資料).
(可 copy the codes，下次比較快).

## 1.6 Export Data from R (csv, txt & other formats)

1. The most flexible command for exporting data from R is write.table

2. write.table (DataToExport, file = "ExportedFileName.csv", sep = ",") ?csv ⇒ 也可寫 write.csv( , )
   (檔名). (存成什麼檔名). "," or "\t" → .txt 以後面就不用打 sep

3. to get rid of row names while exporting ⇒ row.name = FALSE

4. 如果要把檔案存到別的地方 ⇒ file = "path / filename"

## 1.7 Importing, checking and working with Data in R

[Way 1]

1. Data1 <- read.table(file = "___/___/___", header = TRUE, sep = "\t")

[Way 2]

2. Data2 <- read.table(file.choose( ), header = TRUE, sep = "\t")

[Way 3]

3. Import Dataset → From Excel

4. dim(Data1) : 可看 Data 維度 ⇒ 725 6 ⇒ 代表 725 列. 6 欄

5. head(Data1) : 可看前 6 列

6. tail(Data1) : 可看最後 6 列

7. Data1[c(5,6,7,8,9), ] = Data1[c(5:9), ] : 跑出 5~9 列

8. Data1[-(4:722), ] : 可看 4~722 以外的列

9. names(Data1) : 跑出 variables ⇒ "LungCap" "Age" "Height" "Smoke" "Gender" "Caesarean"

## 1.8 Working with Variables and Data in R

[Way 1]

1. mean(Data1$Age) ⇒ $ ⇒ extract the variable
   (資料名) (variable)

[Way 2]

2. 1° attach(Data1) ⇒ 先把資料 attach 到 R 的 memory, 就可以不用在變數前面加上 $.
   2° mean(Age)

3. detach(Data1)

4. class(Age) : 可看變數的資料類型為何

5. levels(Smoke) ⇒ "Yes" "No" ; levels(Gender) ⇒ "male" "female"

6. summary(Data1) → 跑出統計資料

7. x <- as.factor(x)

eg: x <- c(0,1,1,1,0,0,0,0,0,0)
class(x) ⇒ "numeric"
summary(x)

cont. x <- as.factor(x)
class(x) ⇒ "factor"
summary(x)
⇒ 0 1
7 3 → 代表7字0；3字1.

## 1.9 Subsetting Data in R with square Brackets (Sort / Select)

要算 Data 1 裡面的数據.

1° 先讀進檔案到 R. (read.table( ))

2° attach(Data1)

3°
1. mean(Age[Gender=="female"]) ⇒ 12.44972

2. ① FemData <- Data1[Gender=="female"]
MaleData <- Data1[Gender=="male"]
) ← 將資料分類成2種.

② dim(FemData) ⇒ 358 6
dim(MaleData) ⇒ 367 6

① summary(Gender) ⇒ female male
358 367

3. ① MaleOver15 <- Data1[Gender=="male" & Age>15] ← 要找 age>15 的男生有幾個.
② dim(MaleOver15) ⇒ 89 6

## 1.10 Logic Statements (TRUE/FALSE), cbind & rbind Functions in R

Eg:
① Age[1:5] ⇒ 6 18 16 14 5
② temp <- Age>15 ⇒ FALSE TRUE TRUE FALSE FALSE
[Another]
③ temp2 <- as.numeric(Age>15) 0 1 1 0 0

Eg:
① FemSmoke <- Gender=="female" & Smoke=="yes" ← 可找抽菸女性.
② FemSmoke[1:5] ⇒ FALSE TRUE FALSE FALSE FALSE

Eg: MoreData <- cbind(Data1, FemSmoke) ← 加入新的欄位資料
原本的 要加的

- rm(list=ls()) : 清空 workspace
or
Session → Clear workspace

## 1.11 Setting Up Working directory in R

1. getwd( ) : 找到目前的 working directory

[Way 1]
2. setwd("路徑") : 設一個新的 working directory
   ① projectWD ← "/Users .../ .../"
   ② setwd(projectWD)

[Way 2] 3. Session → set working directory → choose directory

[Way 1]
4. save.image("FirstProject.Rdata")

[Way 2]
5. Session → set workspace as

6. 1. load("FirstProject.Rdata") : 把資料讀進 workspace
   or 2. load(file.choose())
   or 3. Session → load workspace

## 1.12 Writing Scripts in R

1. File → New → R Script
   ↘ Open File

2. Tab : 給 code 建議.

3. save.image("FirstProject.Rdata") : save the workspace image

## 1.13 Installing Packages in R

1. install.packages("epiR")
   or (epiR: package 名稱, if 空白→會跳出一排可選.)
2. Tools → Install Packages → CRAN. 名稱.

3. library("epiR")

4. "R-project.org" website 有完整的 packages

5. remove.package("epiR")

## 1.14 Customizing R studio

1. Tools → Options

## 1.15 Apply Function in R.

1. Apply function are a set of loop functions in R.
2. more effient than a "for loop".
3. apply (X, MARGIN, FUN, ...)
   - MARGIN: 1 = row; 2 = column (跟列 or 欄)
   - FUN: function

要算不同支股票的 mean price.

StockData

| | Stock1 | Stock2 | Stock3 |
|---|---|---|---|
| Day1 | | | |
| Day2 | | | |
| Day3 | | | |
| ⋮ | | | |

apply (X = StockData, Margin = 2, FUN = mean (or MAX.), na.rm = TRUE) ↘ remove N/A.

FUN = quantile, probs = c(0.2, 0.8) → 20%, 80%

( .... FUN = plot, type = "l" ) → produce a line plot

( .... FUN = plot, type = "l", main = "stock", ylab = "Price", xlab = "Day" )

## 1.16 tApply Function in R

1. t-apply can be used to apply a function to subsets of a variable or vector

先讀入資料 → summary ( ) → attach ( )

2. t-apply (X, INDEX, FUN = NULL, ..., simplify = TRUE)
   - INDEX: a grouping variable that is the same length as X and is used to create the subsets of data

7. "by" 跟 "tapply" 功能一樣，只是 by 回傳的是向量.

eg: tapply (X = Age, INDEX = Smoke, FUN = mean, na.rm = T)

3. mean (Age [Smoke == "no"])
   mean (Age [Smoke == "yes"]) → (也可達到一樣的效果)

4. tapply (Age, Smoke, Summary)

5. tapply (X = Age, INDEX = list (Smoke, Gender), FUN = mean, na.rm = T)
   (同時考慮2個條件)

6. mean (Age [Smoke == "no" & Gender == "female"])
   ~ "no" ~ "male"
   ~ "yes" ~ "female"

↗ 效果一樣